A Duplication and Loop Checking Free Proof System for S4

Guido Governatori
CIRFID, University of Bologna
via Galliera, 3
40121 Bologna, Italy
governat@cirfid.unibo.it

1 Introduction

Most of the sequent/tableau based proof systems for the modal logic $S4$ need to duplicate formulas and thus are required to adopt some method of loop checking [7, 13, 10]. In what follows we present a tableau-like proof system for $S4$, based on D'Agostino and Mondadori's classical KE [3], which is free of duplication and loop checking. The key feature of this system (let us call it $KES4$) consists in its use of (i) a label formalism which models the semantics of the modal operators according to the usual conditions for $S4$; and (ii) a label unification scheme which tells us when two labels "denote" the same world in the $S4$-model(s) generated in the course of proof search. Moreover, it uses special closure conditions to check models for putative contradictions.

2 Label Formalism

Let $\Phi_C = \{w_1, w_2, \dots\}$ be a non empty set of constant world symbols, and let $\Phi_V = \{W_1, W_2, \dots\}$ be a non empty set of variable world symbols.

The set $\Im$ is now defined as follows:

$$\begin{aligned} \Im &= \bigcup_{1\leq i} \Im_i \text{ where } \Im_i \text{ is :} \\ &\Im_1 = \Phi_C \cup \Phi_V; \\ &\Im_2 = \Im_1 \times \Phi_C; \\ &\Im_{n+1} = \Im_1 \times \Im_n. \end{aligned}$$

That is a world-label is either (i) an element of the set Φ_C, or (ii) an element of the set Φ_V, or (iii) a path term (k', k) where (iiia) $k' \in \Phi_C \cup \Phi_V$ and (iiib) $k \in \Phi_C$ or $k = (i', i)$ where (i', i) is a label. From now on we shall use $i, j, k, \dots$ to denote arbitrary labels.

According to the above intuitive explanation, we may think of a label $i \in \Phi_C$ as denoting a world (a given one), and a label $i \in \Phi_V$ as denoting a set of worlds (any world) in some L-model. A label $i = (k', k)$ may be viewed as representing a path from k to a (set of) world(s) k' accessible from k.

Example 1. The label (W_1, w_1) represents a path which takes us to the set W_1 of worlds accessible from w_1; $(w_2, (W_1, w_1)))$ represents a path which takes us to a world w_2 accessible via any world accessible from w_1, (i.e., accessible from the sub-path (W_1, w_1)) and so on.

A bit of terminology:

Definition 1. For any label $i = (k', k)$ we call k' the *head* of i, k the *body* of i, and denote them by $h(i)$ and $b(i)$ respectively. Notice that these notions are recursive (they correspond to projection functions): if $b(i)$ denotes the body of i, then $b(b(i))$ will denote the body of $b(i)$, $b(b(b(i)))$ will denote the body of $b(b(i))$; and so on.

For example, if i is $(w_4, (W_3, (w_3, (W_2, w_1))))$, then $b(i) = (W_3, (w_3, (W_2, w_1)))$, $b(b(i)) = (w_3, (W_2, w_1))$, $b(b(b(i))) = (W_2, w_1)$, $b(b(b(b(i)))) = w_1$.

Definition 2.

1. We call each of $b(i)$, $b(b(i))$, etc., a segment of i. Let $s(i)$ denote any segment of i (obviously, by definition every segment $s(i)$ of a label i is a label); then $h(s(i))$ will denote the head of $s(i)$.

2. For any label i, we define the length of i, $l(i)$, as the number of world-symbols in i, i.e. $l(i) = n \Leftrightarrow i \in \Im_n$. $s^n(i)$ will denote the segment of i of length n, i.e. $s^n(i) = s(i)$ such that $l(s(i)) = n$; and i^n will denote $h(s^n(i))$.

3. For any label $i, l(i) > n$, we define the countersegment of $s^n(i)$ (*countersegment-n*, $c^n(i)$, i.e. what remains of i after deleting $s^n(i)$), as:

$$c^n(i) = h(i) \times (\cdots \times (h(s^k(i)) \times (\cdots \times (h(s^{n+1}(i)), w_0))))(n < k < l(i))$$

According to the above definition, given the label $i = (w_4, (W_3, (w_3, (W_2, w_1))))$, its length $l(i)$ is 5, its segment of length 3 is $s^3(i) = (w_3, (W_2, w_1)$, and its countersegment-3 is $c^3(i) = (w_4, (W_3, w_0))$. It is worth noting that w_0, in the countersegment-n of i, is a dummy label, i.e., it is a label which does not appear in the given label; the context in which such a notion is applied will tell us what w_0 stands for.

Definition 3. We shall call a label i *restricted* if $h(i) \in \Phi_C$, otherwise we call it *unrestricted.*

Restricted labels, such as $(w_4, (W_3, w_1))$, represent a given world, namely the world denoted by the heads, whereas an unrestricted label (W_2, w_1) stands for a world accessible from its ancestors, the world(s) denoted by the label(s) belonging to the body.

3 Unifications

Labels are manipulated according to rules —unifications— simulating the accessibility relations of the underlying logics. It is worth noting that we shall use two different notions of unifications, namely σ^L or "high" unification, which is meant to mirror a single constraint on R; while the notion of σ_{S4} or "low" unification (which includes the former), is used to simulate the full accessibility restrictions which hold in $S4$-models. In general both high and low unifications are necessary for modal logics, where we have several accessibility relations acting differently, and each relation has its own high unification, and the various high unifications are combined into the low unification which models such logics.

High unifications: We define a substitution in the usual way as a function

$$\sigma \quad : \quad \Phi_V \longrightarrow \Im$$

For two labels i and k, and a substitution σ we shall use $(i, k)\sigma$ to denote both that i and k are unifiable (briefly, are σ-unifiable), and the result of their unification. On this basis we define several logic-dependent notions of σ-unification. The notions of two labels i, k being σ^L-unifiable, for the the accessibility relations of $S4$ are as follows:

$$(i,k)\sigma^D = (i,k)\sigma$$

$$(i,k)\sigma^T = \begin{cases} (s^{l(k)}(i),k)\sigma & l(i) > l(k), \text{ and} \\ & \forall m \geq l(k), (i^m, h(k))\sigma = (h(i),h(k))\sigma \\ (i, s^{l(i)}(k))\sigma & l(k) > l(i), \text{ and} \\ & \forall m \geq l(i), (h(i), k^m)\sigma = (h(i),h(k))\sigma \end{cases}$$

The labels $(W_3,(w_3,(W_1,(w_2,w_1))))$, $(w_3,(W_2,w_1))$ σ^T-unify on $(w_3,(w_2,w_1))$ because $(W_3,w_3)\sigma = (w_3,w_3)\sigma = (W_1,w_3)\sigma = w_3$ and $((W_1,(w_2,w_1)),(w_3,(W_2,w_1)))\sigma^D$, and $((w_2,w_1),(W_2,w_1))\sigma^D$.

$$(i,k)\sigma^4 = \begin{cases} c^{l(i)}(k) & l(k) > l(i), h(i) \in \Phi_V \text{ and} \\ & w_0 = (i, s^{l(i)}(k))\sigma \\ c^{l(k)}(i) & l(i) > l(k), h(k) \in \Phi_V \text{ and} \\ & w_0 = (s^{l(k)}(i),k)\sigma \end{cases}$$

The above notions are meant to mirror the conditions on R in the various L-models. For the notion of σ^T-unification, take for example the labels $i = (w_3,(W_1,w_1)))$ and $k = (w_3,(W_2,(w_2,w_1)))$. Here $(W_2,w_3)\sigma = (w_3,w_3)\sigma$. Then i and k σ^T-unify to $(w_3,(w_2,w_1))$. This intuitively means that the world w_3, accessible from a sub-path $s(k) = (W_2,(w_2,w_1))$, after the deletion of, is accessible from any path i which turns out to denote the same world(s) as $s(k)$; this is possible because the step from w_2 to W_2 is irrelevant because of the reflexivity relations of the model. For the notion of σ^4-unification, take for example $i = (W_3,(w_2,w_1))$ and $k = (w_5,(w_4,(w_3,(W_2,w_1))))$. Here $s^{l(i)}(k) = (w_3,(W_2,w_1))$. Then i and k σ^4-unify to $(w_5,(w_4,(w_3,(w_2,w_1))))$ since $((W_3,(w_2,w_1)),(w_3,(W_2,w_1)))\sigma$. This intuitively means that all the worlds accessible from a sub-path $s^{l(i)}(k)$ of k are accessible from any path i which leads to the same world(s) denoted by $s(k)$.

Low Unification: We are now able to combine the above high unifications in a single unification, corresponding to the $S4$ accessibility relation.

$$(i,k)\sigma_{S4} = \begin{cases} (c^n(i), c^m(k))\sigma^{DT4} \\ (i,k)\sigma^{DT4} \end{cases}$$

where $w_0 = (s^n(i), s^m(k))\sigma_{S4}$ and

$$(i,k)\sigma^{DT4} = \begin{cases} (i,k)\sigma^D & \text{if } l(i) = l(k) \\ (i,k)\sigma^T & \text{if } h(shortest\{i,k\}) \in \Phi_C \\ (i,k)\sigma^4 & \text{if } h(shortest\{i,k\}) \in \Phi_V \end{cases}$$

The above σ_{S4}-unification is similar to Ohlbach's path-separation and splitting rules, see [16]; however the splitting rule requires some new world variables thus implicitly using duplication.

Properties of Labels and Unifications: We shall provide some useful properties of labels and unifications.

Fact 1.

- *If* $i, k \in \Phi_C$ *and* $(i,k)\sigma = l$ *then* $i = k = l$;

- *If $i \in \Phi_C$ and $k \in \Phi_V$ then $(i,k)\sigma = i$;*
- *If i,k are unrestricted and $(i,k)\sigma = l$ then also l is unrestricted;*
- $\forall i \in \Im\ (i,i)\sigma = i$.

Theorem 2. *If $(i,k)\sigma_{S4} = l$ then $(i,l)\sigma_{S4}$ and $(l,k)\sigma_{S4}$.*

Proof. The notion of σ_{S4} is recursive, therefore we can prove the theorem by induction on the number of applications of σ^{DT4} occurring in a σ_{S4}-unification.

Let n be the number of applications of σ^{DT4} in a σ_{S4}-unification.

If $n = 1$ then we prove

$$(i,k)\sigma^{DT4} = l \Rightarrow (i,l)\sigma^{DT4}, (k,l)\sigma^{DT4} \tag{1}$$

i.e. we have to prove the property for σ^{DT4}.[1]

$$(i,k)\sigma^{DT4} = \begin{cases} (i,k)\sigma^D & l(i) = l(k) \\ (i,k)\sigma^T & l(i) < l(k), h(i) \in \Phi_C \\ (i,k)\sigma^4 & l(i) < l(k), h(i) \in \Phi_V \end{cases}$$

At this point we prove the property stated in 1 by induction on the length of labels.

If $\min\{l(i), l(k)\} = 1$ then we assume that $l(i) = 1$ (the proof when $l(k) = 1$ is analogous). If $i \in \Phi_C$ then:

If also $l(k) = 1$, we apply σ^D; in every case $l = (i,k)\sigma^D = i$, see fact 1, but $(i,i)\sigma^D$ and $(i,k)\sigma^D$.

If $l(k) > 1$ then $(i,k)\sigma^T$, but $l = (i,k)\sigma^T = (i, s^1(k))\sigma^T = i$, therefore $(i,i)\sigma^D$ and $(i,k)\sigma^T$.

If $i \in \Phi_V$ then by the definition of σ it unifies with all the labels, and in particular $(i,k)\sigma^D = k = l$, therefore $(i,k)\sigma^D$ and $(k,k)\sigma^D$.

Let us suppose now that $\min\{l(i), l(k)\} = n > 1$, and that the property holds up to n for σ^{DT4}.

If $l(i) = l(k)$ then $(i,k)\sigma^D = l$, by inductive hypothesis $(b(i), b(l))\sigma^D$, $(b(k), b(l))\sigma^D$, $(h(i), h(l))\sigma^D$, and $(h(k), h(l))\sigma^D$; therefore $(i,l)\sigma^D$ and $(k,l)\sigma^D$.

If $l(i) < l(k)$ and $h(i) \in \Phi_C$ then $(i,k)\sigma^D = l$, by inductive hypothesis $(b(i), b(l))\sigma^D$, $(b(k), b(l))\sigma^D$; by the definition of σ^T, we know that $l^n = (h(i), h(k))\sigma = (h(i), k^{l(i)})\sigma)$; therefore $(i,l)\sigma^D$ and $(k,l)\sigma^T$.

If $l(i) < l(k)$ and $h(i) \in \Phi_V$ then $(i,k)\sigma^4 = c^{l(i)}(k)$ where $w_0 = (i, s^{l(i)}(k))\sigma$; by inductive hypothesis and the definition of σ we have $(i, s^{l(i)}(l))\sigma$ and $(s^{l(i)}(k), s^{l(i)}(l))\sigma$ and therefore $(i,l)\sigma^4$ and $(k,l)\sigma^D$.

We have thus proved the inductive base for the theorem.

We can now assume that the theorem holds up to the n-th application of σ^{DT4}; by the definition of σ_{S4}, $(s^n(i), s^m(k))\sigma_{S4} = w_0 = s^l(l)$ and $(c^n(i), c^m(k)j)\sigma^{DT4} = c^l(l)$, but, by inductive hypothesis $(s^n(i), s^l(l))\sigma_{S4}$ and $(s^m(k), s^l(l))\sigma_{S4}$; by the property we have just proved for σ^{DT4} $(c^n(i), c^l(l))\sigma^{DT4}$ and $(c^m(k), c^l(l))\sigma^{DT4}$, which implies $(i,l)\sigma_{S4}$ and $(k,l)\sigma_{S4}$ $\square$

[1] Hereafter, in order to shorten proofs, when we have to consider labels of different lengths, we shall assume, unless specified, the first to be the shorter; however, proofs for the opposite cases carry out in the same way.

4 Inference Rules

The rules of $KES4$ will be defined for LS-formulas, where an LS-formula (labelled signed formula) is an expression of the form:

$$X, i$$

where X is a signed formula [17] and $i \in \Im$.

Definition 4. Two LS-formulas X, i and Z, k such that $Z = X^C$ will be called *complementary*; moreover $T\Diamond\neg A, F\Box\neg A, F\Diamond\neg A$ and $T\Box\neg A$ are also, respectively, complementary to $T\Box A, F\Diamond A, F\Box A$ and $T\Diamond A$. Given two complementary formulas X, i and Z, k, if $(i, k)\sigma_{S4}$ they will be called σ_{S4}-*complementary.*

$KES4$ has the following inference rules:

$$\frac{\alpha, i}{\alpha_1, i} \qquad\qquad \frac{\alpha, i}{\alpha_2, i} \qquad\qquad (\alpha\text{-rules})$$

$$\frac{\begin{array}{c}\beta, i\\ \beta_1^C, k\end{array}}{\beta_2, (i,k)\sigma_{S4}}[(i,k)\sigma_{S4}] \qquad\qquad \frac{\begin{array}{c}\beta, i\\ \beta_2^C, k\end{array}}{\beta_1, (i,k)\sigma_{S4}}[(i,k)\sigma_{S4}] \qquad\qquad (\beta\text{-rules})$$

$$\frac{\nu, i}{\nu_0, (i', i)}[i' \in \Phi_V \text{and new}] \qquad\qquad (\nu\text{-rules})$$

$$\frac{\pi, i}{\pi_0, (i', i)}[i' \in \Phi_C \text{ and new}] \qquad\qquad (\pi\text{-rules})$$

$$\frac{}{X, i \qquad X^C, i}[i\text{restricted}] \qquad\qquad (PB)$$

$$\frac{\begin{array}{c}X, i\\ X^C, k\end{array}}{\times(i,k)\sigma_{S4}}[(i,k)\sigma_{S4}] \qquad\qquad (PNC)$$

Here the α-rules are just the usual linear branch-expansion rules of the tableau method, while the β-rules correspond to such common natural inference patterns as *modus ponens, modus tollens*, etc.

The rules for the modal operators bear a not unexpected resemblance to the familiar quantifier rules of the tableau method. "i' new" in the proviso for the ν- and π-rule obviously means: i' must not have occurred in any label yet used.

Notice that in all inferences via an α-rule the label of the premise carries over unchanged to the conclusion, and in all inferences via a β-rule the labels of the premises must be σ_{S4}-unifiable, so that the conclusion inherits their unification.

PB (the "Principle of Bivalence") represents the (LS-version of the) semantic counterpart of the cut rule of the sequent calculus (intuitive meaning: a formula A is either true or false in any given world).

PNC (the "Principle of Non-Contradiction") corresponds to the familiar branch-closure rule of the tableau method, which says that from a contradiction of the form (the occurrence of a pair of σ_{S4}-complementary LS-formulas), X, i and X^C, k on a branch, we may infer the closure of the branch. The $(i, k)\sigma_{S4}$ in the "conclusion" of PNC means that the contradiction holds "in the same world".

5 Proof Search

As usual with refutation methods, a proof of a formula A of $S4$ consists in attempting to construct a counter-model for A by assuming that A is false in some arbitrary $S4$-model. Every successful proof discovers a contradiction in the putative counter-model. In this section we describe an algorithm which does this job.

In what follows by a $KES4$-*tree* we shall mean a tree generated by the inference rules of $KES4$.

Definition 5. A branch τ of a $KES4$-tree will be said to be σ_{S4}-*closed* if it ends with an application of PNC. A $KES4$-tree $\mathcal{T}$ will be said to be σ_{S4}-*closed* if all its branches are σ_{S4}-closed. By an $KES4$-*proof* of a formula A we shall mean a σ_{S4}-closed $KES4$-tree starting with FA, i. Finally, a formula A is a $KES4$-*theorem* ($\vdash_{KES4} A$) if there exists a $KES4$-proof of A.

Definition 6.

- Each formula depends on itself;
- a formula B depends on A either if it is obtained through an application of the α-, ν- or π-rules, or it is obtained through an application of $KES4$'s rules on formulas depending on A;
- a formula C depends on A, B if it is obtained through an application of a β-rule where A, B are its premises;
- if C depends on A, B then C depends on A ,and C depends on B.

Definition 7. Given a branch τ of a $KES4$-tree, we shall call an LS-formula X, i *E-analysed in* τ if either:

1. X is of type α and both α_1, i and α_2, i occur in τ; or
2. X is of type β and one of the following conditions is satisfied:
   (a) if β_1^C, k not depending on β occurs in τ and $(i,k)\sigma_{S4}$, then also $\beta_2, (i,k)\sigma_{S4}$ occurs in τ,
   (b) if β_2^C, k not depending on β occurs in τ and $(i,k)\sigma_{S4}$, then also $\beta_1, (i,k)\sigma_{S4}$ occurs in τ; or
3. X is of type ν and $\nu_0, (i', i)$ occurs in τ for some $i' \in \Phi_V$ not previously occurring in τ, or
4. X is of type π and $\pi_0, (i', i)$ occurs in τ for some $i' \in \Phi_C$ not previously occurring in τ.

Definition 8. Given a branch τ of a $KES4$-tree, we shall call a β-formula β, i *fulfilled in* τ if there exists a label k in τ such that $(i, K)\sigma_{S4}$ and either β_1, k or β_2, k is in τ. We shall say that an LS-formula of type β is analysed in a branch τ if it is either E-analysed or fulfilled.

Definition 9. We shall call a branch τ of a $KES4$-tree *E-completed* if every LS-formula in it is E-analysed. We shall say a branch τ of a $KES4$-tree *completed* if it is E-completed and all the pair of complementary LS-formulas are not σ_{S4}-complementary. We shall call a $KES4$-tree *completed* if every branch is completed.

The procedure for $KES4$-trees starts from the 1-branch, 1-node tree consisting of FA, i, where i is a restricted label, and it applies the rules of $KES4$ until the resulting $KES4$-tree is either closed or completed. At each stage of the proof search

(i). we choose an open non-completed branch τ. If τ is not E-completed, then

(ii). we apply the 1-premise rules;

(iii). we apply the 2-premise rules;

(iv). we choose an LS-formula of type β which is not yet analysed in the branch and we apply PB so that the resulting LS-formulas are β_1, i' and β_1^C, i' (or, equivalently β_2, i' and β_2^C, i'), where $i = i'$, if i is restricted; otherwise i' is a restricted label unifying with i already occurring in the branch; otherwise i' is obtained from i by instantiating $h(i)$ to a constant not occurring in τ;

(v). if the branch is E-completed and it contains complementary formulas which may be σ_{S4}-complementary, then we have to see whether a restricted label unifying with both the labels of the complementary formulas occurs previously in the branch; if such a label exists or can be built using already existing labels and unification rules, then the branch is closed; let $\mathcal{L}$ be the set of labels occurring in a branch, a label k can be built if: $\exists i, j \in \mathcal{L}$ such that $k = c^n(i)$ where $w_0 = j$ if

   (a) $(s^n(i), j)\sigma_L$, and

   (b) j is the label of a formula of type β such that PB has been applied to.

Notice that the applications of PB fulfil the β-formulas PB is applied to.

Loop checking plays a prominent role in ensuring the termination of proofs [7, 13, 10], this problem is due to the duplication implied by the transitivity axiom; $KES4$ avoids useless duplications of formulas since the modal operators are analysed only once, and the information they give are encoded in the labels.

6 Example

In this section we present $KES4$-proofs both of the characteristic axioms of $S4$, and of some other formulas, to show $KES4$'s features. What is important to note is that the proofs follow the procedure of section 5.

The following is a proof of the formula $\Box(A \to B) \to (\Box A \to \Box B)$.

1. $F\Box(A \to B) \to (\Box A \to \Box B)$	w_1
2. $T\Box(A \to B)$	w_1
3. $F\Box A \to \Box B$	w_1
4. $T(A \to B)$	(W_1, w_1)
5. $T\Box A$	w_1
6. $F\Box B$	w_1
7. TA	(W_2, w_1)
8. FB	(w_2, w_1)
9. TB	(W_2, w_1)
10. $\times$	(w_2, w_1)

Here the steps from (1) to (8) have been obtained from 1-premise rules; step (9) has been obtained through an application of a β-rule on (4) and (7); σ_{S4}-closure, step (10), follows from the fact that (W_2, w_1) and (w_2, w_1) σ^D-unify.

The following is a proof of the formula $\Box A \to A$.

1. $F\Box A \rightarrow A$	w_1
2. $T\Box A$	w_1
3. FA	w_1
4. TA	(W_1, w_1)
5. $\times$	w_1

Here σ_{S4}-closure follows from (3) and (4) which are σ_{S4}-complementary (their labels are σ^T-unifiable, being $(w_1, w_1)\sigma = (w_1, W_1)\sigma$).

The following is a proof of the formula $\Box A \rightarrow \Box\Box A$.

1. $F\Box A \rightarrow \Box\Box A$	w_1
2. $T\Box A$	w_1
3. $F\Box\Box A$	w_1
4. TA	(W_1, w_1)
5. $F\Box A$	(w_2, w_1)
6. FA	$(w_3, (w_2, w_1))$
7. $\times$	$(w_3, (w_2, w_1))$

σ_{S4}-closure follows from (4) and (6) which are σ_{S4}-complementary (their labels are σ_{S4}-unifiable since the head of the shorter, (W_1, w_1), is a variable and $((W_1, w_1), (w_2, w_1))\sigma$).

In the following proof we show the use of a more complex σ_{S4}-unification.

1. $F\Diamond\Box(A \rightarrow \Box\Diamond A)$	w_1
2. $F\Box(A \rightarrow \Box\Diamond A)$	(W_1, w_1)
3. $FA \rightarrow \Box\Diamond A$	$(w_2, (W_1, w_1))$
4. TA	$(w_2, (W_1, w_1))$
5. $F\Box\Diamond A$	$(w_2, (W_1, w_1))$
6. $F\Diamond A$	$(w_3, (w_2, (W_1, w_1)))$
7. FA	$(W_2, (w_3, (w_2, (W_1, w_1))))$
8. $\times$	$(w_2, (w_3, (w_2, (W_1, w_1))))$

Here the labels $(w_2, (W_1, w_1))$ and $(W_2, (w_3, (w_2, (W_1, w_1))))$ σ_{S4}-unify because

$$w_0 = ((W_1, w_1), (w_3, (w_2, (W_1, w_1))))\sigma_{S4}$$

since the two labels σ^4-unify and $(W_2, w_2)\sigma^D$. Therefore $(w_2, (w_3, (w_2, (W_1, w_1))))$ is the label of the closure.

The constraint which prevents the application of a β-rule with respect to a β-formula as a major premise and a formula depending on it as a minor premise, together with the fact that no formula is analysed more than once, prevents both duplications and loops. By way of an example we try to $KES4$-refute the Löb axiom.

1. $F\Box(\Box A \rightarrow A) \rightarrow \Box A$	w_1
2. $T\Box(\Box A \rightarrow A)$	w_1
3. $F\Box A$	w_1
4. $T\Box A \rightarrow A$	(W_1, w_1)
5. FA	(w_2, w_1)
6. $F\Box A$	(w_2, w_1)
7. FA	$(w_3, (w_2, w_1))$

Notice that if the restriction did not hold, we could have applied the β-rule to (4) and (7), thus obtaining $F\Box A, (w_3, (w_2, w_1))$ which implies, by an application of the ν-rule, $FA, (w_4, (w_3, (w_2, w_1)))$, and therefore we must have applied the β-rule again, thus obtaining $F\Box A, (w_4, (w_3, (w_2, w_1)))$ and so on.

In the next $KES4$-tree we can see how σ_{S4}-complementarity is detected.

1. $F\Diamond((\Box(A \vee \Diamond\neg B) \vee \Box B) \wedge (C \vee D)) \vee \Diamond\neg C$	w_1
2. $F\Diamond((\Box(A \vee \Diamond\neg B) \vee \Box B) \wedge (C \vee D))$	w_1
3. $F\Diamond\neg C$	w_1
4. $F(\Box(A \vee \Diamond\neg B) \vee \Box B) \wedge (C \vee D)$	(W_2, w_1)
5. TC	(W_3, w_1)

6. $T\Box(A \vee \Diamond\neg B) \vee \Box B$	w_1	7. $F\Box(A \vee \Diamond\neg B) \vee \Box B$	w_1
8. $FC \vee D$	w_1	11. $F\Box(A \vee \Diamond\neg B)$	w_1
9. FC	w_1	12. $F\Box B$	w_1
10. $\times$		13. $FA \vee \Diamond\neg B$	(w_2, w_1)
		14. FB	(w_3, w_1)
		15. $F\Diamond\neg B$	(w_2, w_1)
		16. TB	$(W_4, (w_2, w_1))$
		17. $\times$	$(w_3, (w_2, w_1))$

Here the tree is closed because, in the right branch, there are two complementary formulas, and we discovered a restricted label which σ_{S4}-unifies with both the labels of the complementary formulas. The labels of 14 and 16 (let us call them i and j respectivelly) do not unify; however we have the label (W_2, w_1), which is a label of a formula PB has been applied to, and it σ_{S4}-unifies with $s^1(i) = w_1$. Thus we can replace w_1 in i with (W_2, w_1), obtaining thus $(w_3, (W_2, w_1))$, which σ_{S4}-unifies with both the labels of the complementary formulas.

In [1] we proposed a different approach with a different σ_{S4}-unification built upon σ^D-, σ^T- and σ^4-unifications but alone it was not sufficient to capture the whole transitivity. It was able to deal with “forward” transitivity, i.e. what happens before a given unrestricted label; however, “backward” transitivity was treated by another operation on labels (reduction) which cuts superfluous labels from a given one. Unfortunately the reduction rule, sometimes, had to be applied more than once with respect to a given label, which amounts to an implicit use of duplication[2]. The system of the present work avoids such a use.

7 Soundness, Completeness and Termination

Let $\mathcal{M} = \langle \mathcal{W}, R, v \rangle$ be an $S4$-model where $\mathcal{W} = \Phi_C$ and Φ_V is interpreted as a subset of $\wp(\mathcal{W})$; R is a transitive and reflexive relation on $\mathcal{W}$; v is as usual.

Let g be a function from $\Im$ to $\mathcal{W}$ thus defined:

$$g(i) = \begin{cases} h(i) & \text{if } h(i) \in \Phi_C \\ h(i) = \{w_i \in \mathcal{W} : g(b(i))Rw_i\} & \text{if } h(i) \in \Phi_V \\ i = \mathcal{W} & \text{if } i \in \Phi_V \end{cases}$$

Let r be a function from $\Im$ to R thus defined:

$$r(i) = \begin{cases} \emptyset & \text{if } l(i) = 1 \\ g(i^1)Rg(i^2), \ldots, g(i^{n-1})Rg(h(i)) & \text{if } l(i) = n > 1 \end{cases}$$

Let f be a function from LS-formulas to v thus defined:

$$f(SA, i) = v(A, g(i)) = S$$

Lemma 3. *For any $i, k \in \Im$ if $(i,k)\sigma_{S4}$ then $g(i) \cap g(k) \neq \emptyset$*

[2]This fact has been pointed out by Ugo Moscato; however the multiple application of reduction were embedded in a single step unification, and strictly speaking it was not a duplication of formulas.

Proof. We prove the lemma by induction on the number of applications of σ^{DT4} in σ_{S4}. To this end, we need first to prove the following:

Lemma 4. *For any $i, k \in \Im$ if $(i, k)\sigma^{DT4}$ then $g(i) \cap g(k) \neq \emptyset$*

Proof. We prove the lemma by induction on the length of labels. If $\min\{l(i), l(k)\} = 1$, then at least one of i and k is either a constant or a variable, so that five cases will be present. By the definition of unifications i, k are either:

(i). two constants, or

(ii). a variable and a constant, or

(iii). two variables, or

(iv). a variable and a label, or

(v). a constant and a label.[3]

Case i) Two constants unify if and only if they are the same constant, and so $i = k$; therefore from definition of g, $g(i) = g(k)$ and so $g(i) \cap g(k) \neq \emptyset$.

Case ii) If i (resp. k) is a variable and k (resp. i) is a constant, then $g(i) \in \mathcal{W}$ and $g(k) = \mathcal{W}$ therefore also in this case $g(i) \cap g(k) \neq \emptyset$.

Case iii) and iv) These cases are identical to the previous ones because: 1) $\mathcal{W}$ is not empty, and 2) the variable is mapped to $\mathcal{W}$ and the label to some world(s) in it.

Case v) This case implies that $(i, k)\sigma^T$. Let us assume, for the sake of economy, that $l(i) = 1$ and $l(k) = n > 1$. If $(i, k)\sigma^L$ then each $h(s(k))$ so that $l(s(k)) > 1$ either belongs to Φ_V, or $h(s(k)) = i$; therefore

$$r(k) = iRk^2, \ldots, k^{n-1}Rk^n \ .$$

If $k^2 \in \Phi_V$ then k^2 denotes the set of worlds accessible from i, but, through reflexivity $i \in k^2$, so we take i as the representant of the set denoted by k^2, but this implies iRk^3. We can repeat the same argument until we arrive at iRk^n; if $k^n \in \Phi_C$ then $i = k^n$ and so they denote the same world; if $k^n \in \Phi_V$ then it denotes the set of worlds accessible from i; but i belongs to such a set, therefore, in all cases $g(i) \cap g(k) \neq \emptyset$.

For the inductive step we have $\min\{l(i), l(k)\} = n > 1$. Let us assume inductively that the lemma is valid up to n; if $l(i) = l(k)$ we shall write i and k as $(h(i), b(i))$ and $(h(k), b(k))$, respectively. Given that $(i, k)\sigma^D$, by the definition of σ^D we get $(b(i), b(k))\sigma^D$, for which the lemma holds; let Γ be one of the worlds shared by $b(i)$ and $b(k)$, whence $\Gamma R h(i)$ and $\Gamma R h(k)$. We have now to analyse only what kind of labels are $h(i)$ and $h(k)$, which falls under the cases i), ii), and iii). Cases i) and ii) are the same as the inductive base; whereas we still have to examine case iii). Both $h(i)$ and $h(k)$ denotes the set of worlds accessible from Γ, but such a set is not empty because of the seriality condition of $\mathcal{M}$.

If $l(i) \neq l(k)$, we shall assume that $l(i) < l(k)$ (the case $l(k) < l(i)$ is dealt with in the same way); if $h(i) \in \Phi_C$ then $(i, k)\sigma^T$ which means $(i, s^{l(i)}(k))\sigma^D$, therefore, combining the proofs of the previous case and case v) of the inductive base we obtain the desired result.

If $h(i) \in \Phi_V$ then $(i, k)\sigma^4$ which means $(b(i), s^{l(i)-1}(k))\sigma^D$, for which the inductive hypothesis holds. Let Γ be such a shared world. We have that $h(i)$ denotes all the worlds accessible from Γ, but, due to transitivity, the world(s) denoted by $h(k)$ belong(s) to $h(i)$ and so $g(i) \cap g(k) \neq \emptyset$. $\square$

[3] Cases ii), iii), and iv) are not found in $KES4$ proofs, but they are useful both for dealing with cases in the inductive step and for case v).

We return now to the proof of the main lemma. σ_{S4} consists of a single step of σ^{DT4}, then $(i,k)\sigma_{S4} = (i,k)\sigma^{DT4}$; by lemma 4 we obtain $g(i) \cap g(k) \neq \emptyset$.

Let us assume, inductively, that the lemma holds up to n. If σ_{S4} consists of $n+1$ σ^{DT4}-unifications, $(i,k)\sigma_{S4} = (c^i(i), c^k(k))\sigma^{DT4}$ where $(s^i(i), s^k(k))\sigma_{S4}$, which contains n applications of σ^{DT4}, and the lemma holds for it. We can now repeat the argument of lemma 4 with respect to $(c^i(i), c^k(k))\sigma^{DT4}$, proving thus that $g(i) \cap g(k) \neq \emptyset$. $\square$

Lemma 5. *For any $i, k \in \Im$ if $f(X,i)$ and $(i,k)\sigma_{S4}$ then $f(X,(i,k)\sigma_{S4})$.*

Proof. Let us suppose that the lemma does not hold, therefore $v(A, g(i)) = S$ and $v(A, g((k,i)\sigma_{S4})) = S^C$. However, according to lemma 3 $g(i) \cap g(k) \neq \emptyset$, which means that there is a world w_n such that $v(A, w_n) = S$ and $v(A, w_n) = S^C$, thus obtaining a contradiction. $\square$

Theorem 6. $\models_{S4} A \iff \vdash_{S4} A$

Proof. For the proof see, for example, [11]. $\square$

Theorem 7. $\vdash_{S4} A \Rightarrow \vdash_{KES4} A$.

Proof. The characteristic axioms of $S4$ and *modus ponens* are provable in $KES4$ (see section 6, and [3] for a proof that *modus ponens* is a derived rule in KE, the propositional subsystem of $KES4$). We give a $KES4$-proof of the rule of necessitation. Let us assume that $\vdash_{KES4} A$. $\Box A$ is proved in $KES4$ as follows.

$$\begin{array}{lc} 1.\ F\Box A & w_1 \\ 2.\ FA & (w_2, w_1) \\ 3.\ \times & (w_2, w_1) \end{array}$$

$\square$

Theorem 8. $\vdash_{KES4} A \Rightarrow \models_{S4} A$

Proof. The α-rules and PB are obviously sound rules in $\mathcal{M}$. For the β-rules and PNC. By the hypothesis: $(i,k)\sigma_{S4}$, then, by lemma 2, $(i,(i,k)\sigma_{S4})\sigma_{S4}$ and $(k,(i,k)\sigma_{S4})\sigma_{S4}$ hence, by lemma 5, the formulas involved have the same value in $g(i)$, $g(k)$ and $g((i,k)\sigma_{S4})$; after that these rules become rules of KE, and thus they are sound rules in $\mathcal{M}$. For ν-rules: let us suppose $\nu = T\Box A$; if we put $g(i) = \Gamma$ and $g((i',i)) = \Gamma'$, then $v(\Box A, \Gamma) = T$; but $v(\Box A, \Gamma) = T \Leftrightarrow \forall \Gamma' : \Gamma R \Gamma', v(A, \Gamma') = T$, and $(\forall \Gamma' : \Gamma R \Gamma', v(A, \Gamma') = T) = f(\nu_0, i')$ with i' unrestricted. The proof for π-rules is similar. $\square$

From theorems 6, 7, and 8 we obtain:

Theorem 9. $\vdash_{KES4} A \iff \models_{S4} A$

In the rest of the section we prove that $KES4$-tree always terminates, since for each formula there are a finite number of sub-formulas and the number of labels which can occur in the $KES4$-tree for a formula A (of L) is limited by the number of modal operators belonging to A.

We shall define the complexity of an LS-formula as the number of logical symbols occurring in it.

Theorem 10. *A $KES4$-tree always terminates*

Proof. We show that each step produces at most a finite number of new LS-formulas, where with *new* we mean that the label has not been previously used with the S-formula.

The procedure for $KES4$-trees stops either when

1. there are no LS-formulas whose complexity is greater than 1, or
2. all the pairs of complementary formulas in a branch are not σ_{S4}-complementary.

We prove the theorem by induction on the length of $KES4$-trees, where with length we mean the number of times the procedure for $KES4$-trees has been applied.

At step 1, α-rules produce two new LS-formulas of less complexity, whereas ν-, and π-rules produce a new LS-formula of less complexity.

At the n-th step α-rule produces at most 2 new LS-formulas of less complexity; and both ν- and π-rules produce a new LS-formulas of less complexity; the β-rules produce at most m new LS-formulas of less complexity, where m is the number of LS-formulas which are the conjugate of an immediate sub-formula of a β-formula, and whose labels σ_{S4}-unify with the label of the β-formula; by induction m is finite. PB produces 2 branches containing a new LS-formulas of less complexity.

If there are some complementary formulas which are not σ_{S4}-complementary, modal PB controls whether a restricted label which σ_{S4}-unifies with both the labels of the complementary formulas occurs in the tree or such a label can be built. But the number of the restricted labels occurring in the tree is finite, since at most it is equal to the number of the LS-formulas occurring in the tree which is finite, and the number or labels that can be built is bounded by the maximum $l(i)$ in the branch, which is finite, and the number of labels occurring in the branch which is finite. □

8 Final Remarks

Several other duplication free systems have been proposed [8, 12], although the former [8] has proved to be incorrect [15] and the latter [12] works only with clausal forms [4]. But translation into such forms as are there proposed is not always possible; in fact it requires new propositional letters and in a finite language we may have a formula which contains all the propositional letters, and its translation requires for a new one, although this turns out to be impossible since all the available letters have been used. In [1] we proposed a system similar to the present one which implicitly uses duplications.

The same problem for intuitionistic logic has been solved by Dyckhoff [5], Dyckhoff Pinto [6] and Migliogli Moscato Ornaghi [14].

The issues of efficiency and complexity have not yet been treated: however, the propositional fragment of $KES4$ benefits the results of KE [2], which provides an efficient alternative to standard tableau methods; experiments with the Halpern and Moses' branching formula [9] show that $KES4$ sharply reduces proof lengths both for such a class of formulae, and when the branching formula is embedded in a formula which requires either duplication or loop check.

We proved that our system is duplication and loop free for canonical trees, i.e., when a uniform strategy has been selected; but pathological formulae for uniform strategies can be found [4]: however, since our set of inference rules is fully invertible then we can choose a not uniform strategy in order to improve efficiency.

Aknowledgements

I would like to thank Alberto Artosi for his help, and the anonymous referees for their valuable suggestions.

References

[1] Alberto Artosi and Guido Governatori. Labelled Model Modal Logic. In *Proceedings of CADE-12 Workshop of Automated Model Building* Nancy, 1994: 11–17.

[2] Marcello D'Agostino. *Investigations into the Complexity of some Propositional Calculi.* PhD Thesis Oxford University, 1990.

[3] Marcello D'Agostino and Marco Mondadori. The Taming of the Cut. *Journal of Logic and Computation*,4, 1994: 285–319.

[4] Stephane Demri. Uniform vs not uniform strategies in modal tableaux. *Journal of Applied and Non Classical Logic*, 1995:

[5] Roy Dyckhoff. Contraction-free sequent calculi for intuitionistic logic. *Journal of Symbolic Logic*, 57, 1992: 795–807.

[6] Roy Dyckhoff and Luis Pinto. Loop-free construction of counter-models for intuitionistic propositional logic Symposia Gaussiana, Eds.: Behara/Fritsch/Lintz, Walter de Gruyter, 1995, pp. 225–232

[7] Melvin Fitting. *Proof Methods for Modal and Intuitionistic Logic.* D. Reidel: Dordrecht, 1983.

[8] Roderic A. Girle. Non-looping tableau for S4. In David Basin, Reiner Hähnle, Bertram Fronhöfer, Joachim Posegga and Camilla Schwind. *Tableaux 93*, Max-Plank-Institute, MPI-I-93-213, Marseille, 1993: 77–88.

[9] Joseph Y. Halpern and Yoram Moses. A guide for completeness and complexity for modal logic of knowledge and belief. *Artificial Intelligence*, 54, 1992: 319–379.

[10] G. E. Hughes and M. Creswell. *An Introduction to Modal Logic.* London, Methuen, 1968.

[11] G. E. Hughes and M. Creswell. *A Companion to Modal Logic.* London, Methuen, 1984.

[12] Jörg Hudelmaier. On a contraction free sequent calculus for the modal logic S4. In Krysia Broda, Marcello D'Agostino, Rajeev Goré, Rob Johnson and Steve Reeves eds. *Tableaux 94*, Imperial College, TR-94/5, Abingdon, 1994: 105–112.

[13] Saul A. Kripke. Semantical analysis of modal logics I. Normal modal propositional calculi. *Zeitschrift für Mathematische Logik und Grundlagen des Mathematik*, 9, 1963: 67–96.

[14] Pierangelo Migliogli, Ugo Moscato and Mario Ornaghi How to avoid duplications in refutation systems for intuitionistic and Kuroda logic. In Krysia Broda, Marcello D'Agostino, Rajeev Goré, Rob Johnson and Steve Reeves eds. *Tableaux 94*, Imperial College, TR-94/5, Abingdon, 1994: 169–187.

[15] Pierangelo Migliogli, Ugo Moscato and Mario Ornaghi Refutation systems for propositional modal logics. In Peter Baumgarter, Reiner Hähnle and Joachim Posegga eds. *Theorem proving with analytic tableaux and related methods*, LNAI 918, Springer Verlag, Berlin, 1995: 95–105.

[16] Hans Jürgen Ohlbach. Semantic-based translation methods for modal logic. *Journal of Logic and Computation*, 1, 1991: 691–746.

[17] Raymond Smullyan. *First-Order Logic*. Springer Verlag, Berlin, 1968.